(12) UK Patent Application (19) GB (11) 2 338 381 (13) A

(43) Date of A Publication 15.12.1999

(21) Application No 9812520.6

(22) Date of Filing 10.06.1998

(71) Applicant(s)
Barclays Bank Plc
(Incorporated in the United Kingdom)
54 Lombard Street, LONDON, EC3P 3AH,
United Kingdom

(72) Inventor(s)
David Alexander Taylor
Mark Jonathan Stirland

(74) Agent and/or Address for Service
R G C Jenkins & Co
26 Caxton Street, LONDON, SW1H ORJ,
United Kingdom

(51) INT CL[6]
H04L 9/32

(52) UK CL (Edition Q)
H4P PDCSA
U1S S2209

(56) Documents Cited
EP 0782296 A2 WO 97/41539 A1 WO 97/23972 A1
US 5615268 A US 5602918 A US 5590197 A
Internet Cryptography by R E Smith Pub Addison Wesley 1997 ISBN 0-201-92480-3 pp 113-116 262-265

(58) Field of Search
UK CL (Edition P) H4P PDCSA PDCSC
INT CL[6] H04L 9/32
Online:WPI,EPODOC,PAJ

(54) Abstract Title
Cryptographic authentication for internet using two servers

(57) In a system for the authentication of transactions over a public network (100), a terminal (14) sends digitally signed transaction data (SD) to a service provider (20) over the public network (100), together with card application data (CAD) generated by a smart card (18). The card application data (CAD) is sent to an authorization server (30) which checks that the smart card (18) is valid and that the card application data (CAD) must have been generated by that smart card (18) in the current transaction. User identification information (ID) is also sent from the terminal (14) to the service provider (20) and thence to the authorisation server (30), where this information (ID) is checked against the correct user details for the smart card (18). The results of these checks are indicated in a digitally signed authorisation response (ARES) from the authorization server (30) to the service provider (20), which then determines whether to proceed with the transaction by setting acceptance criteria for the current transaction and determining from the authorisation response (ARES) whether these criteria are met.

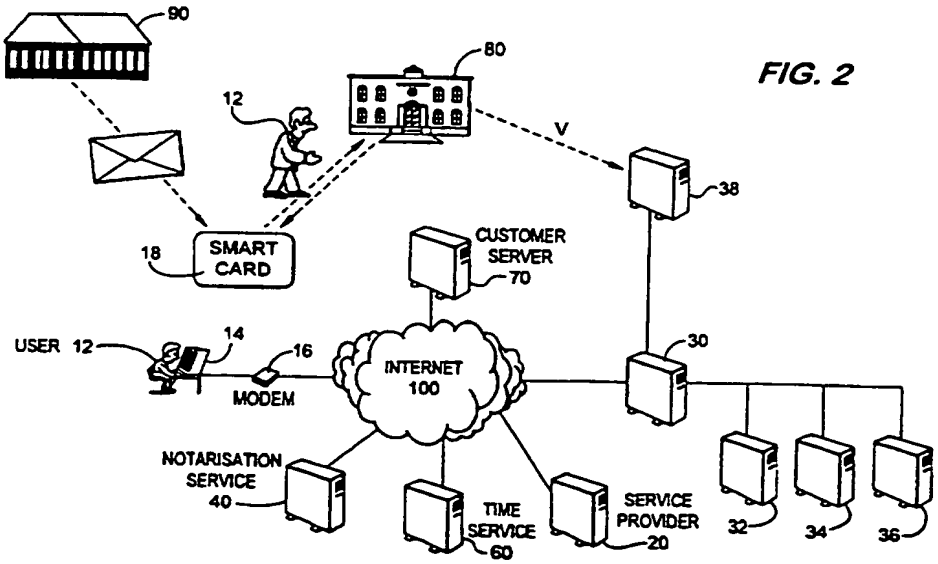


FIG. 2

GB 2 338 381 A

At least one drawing originally filed was informal and the print reproduced here is taken from a later filed formal copy.

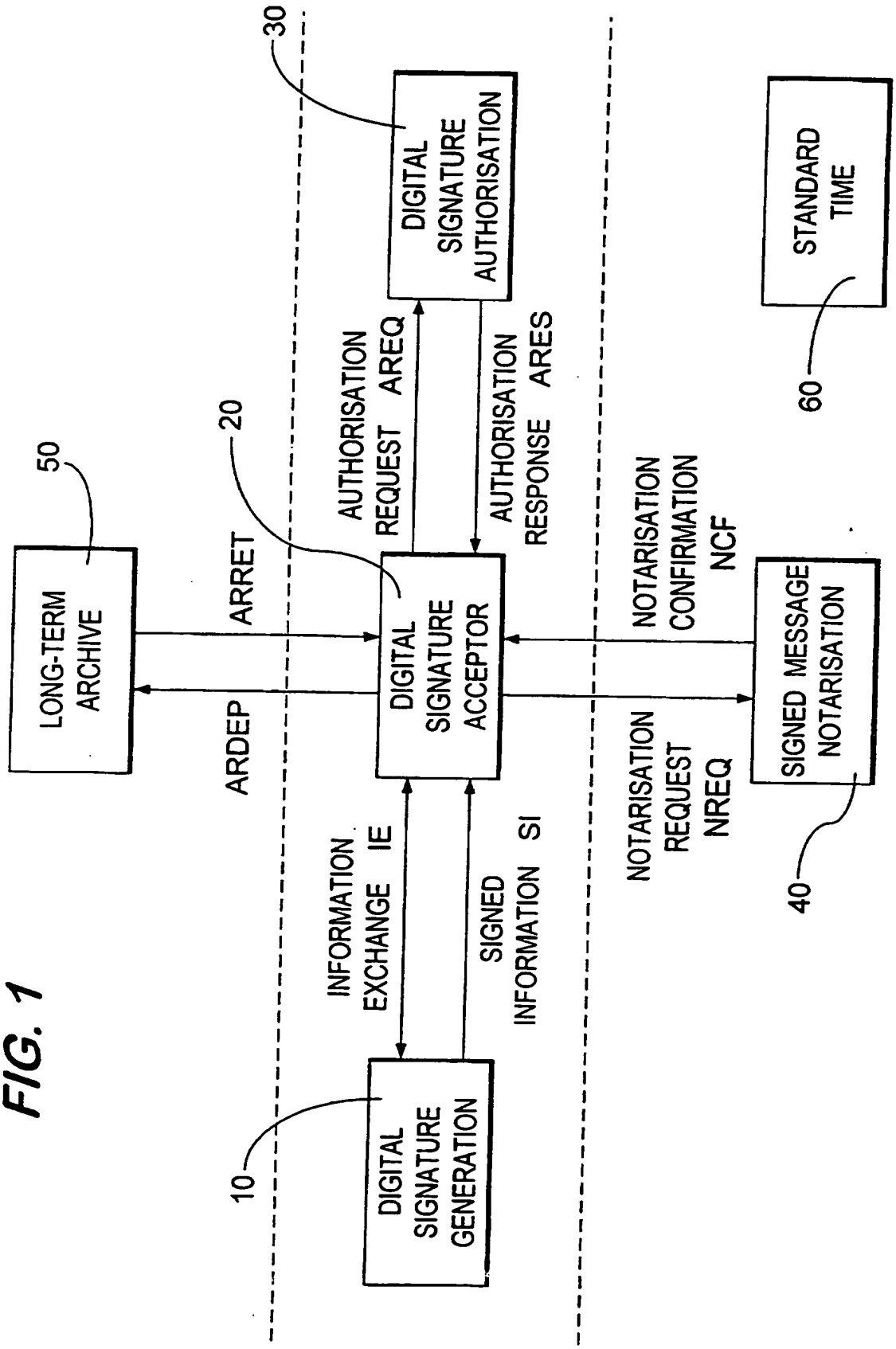
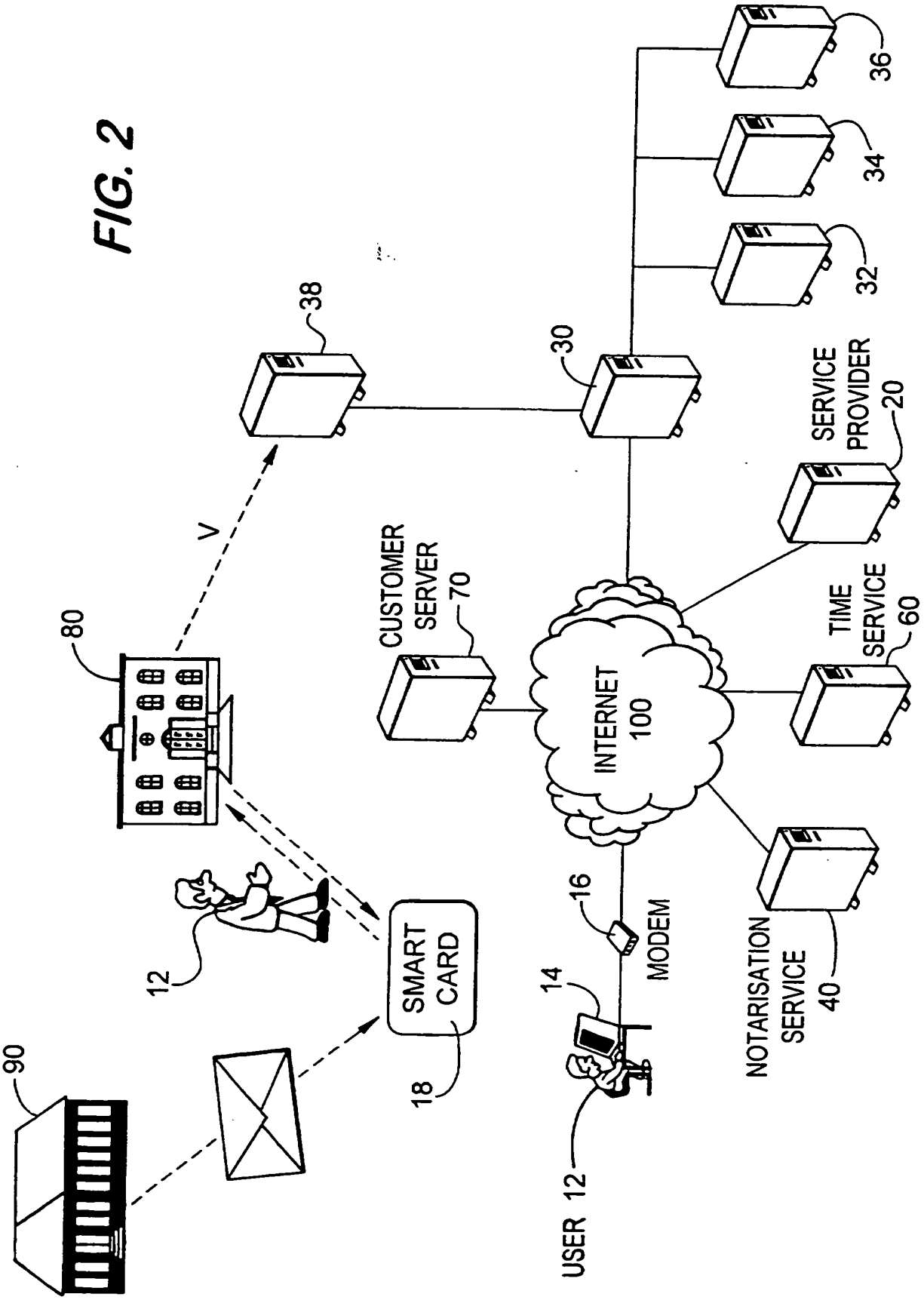


FIG. 1

FIG. 2



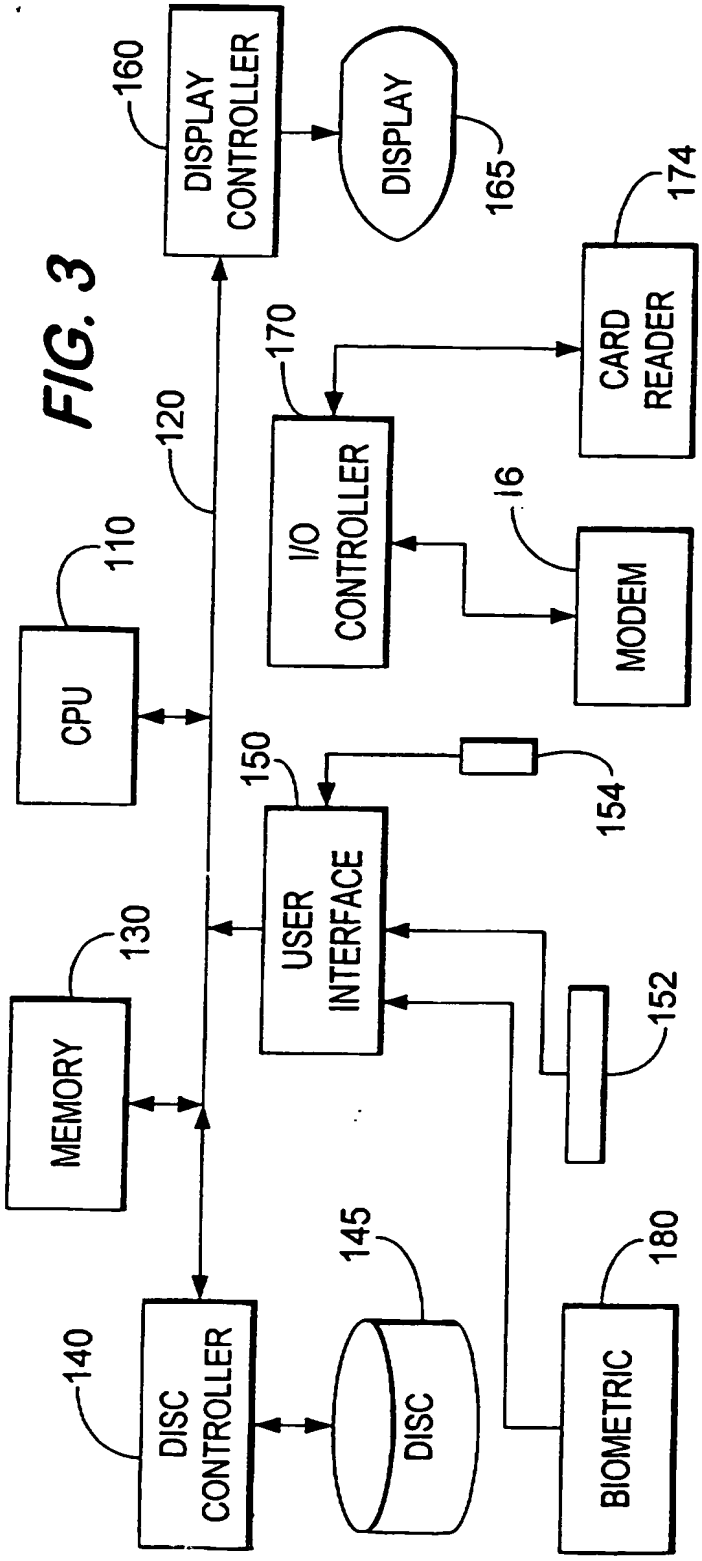


FIG. 3

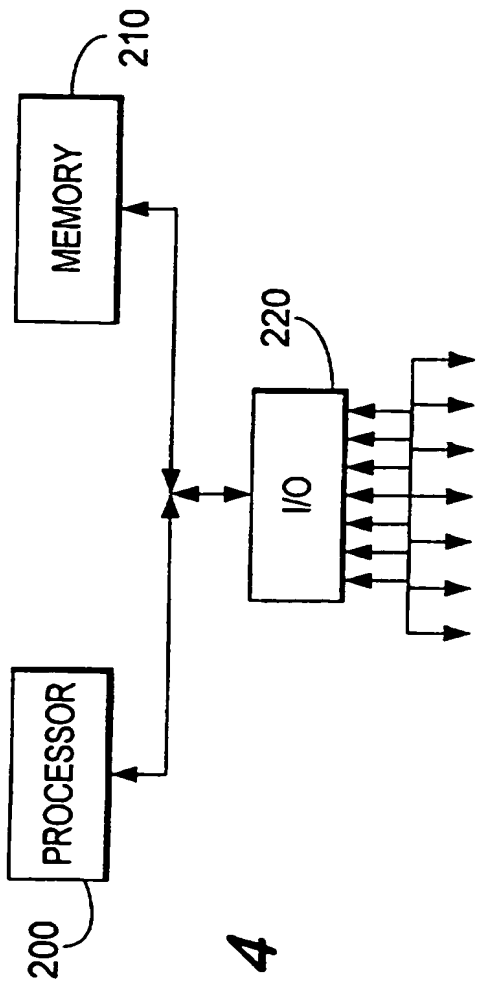


FIG. 4

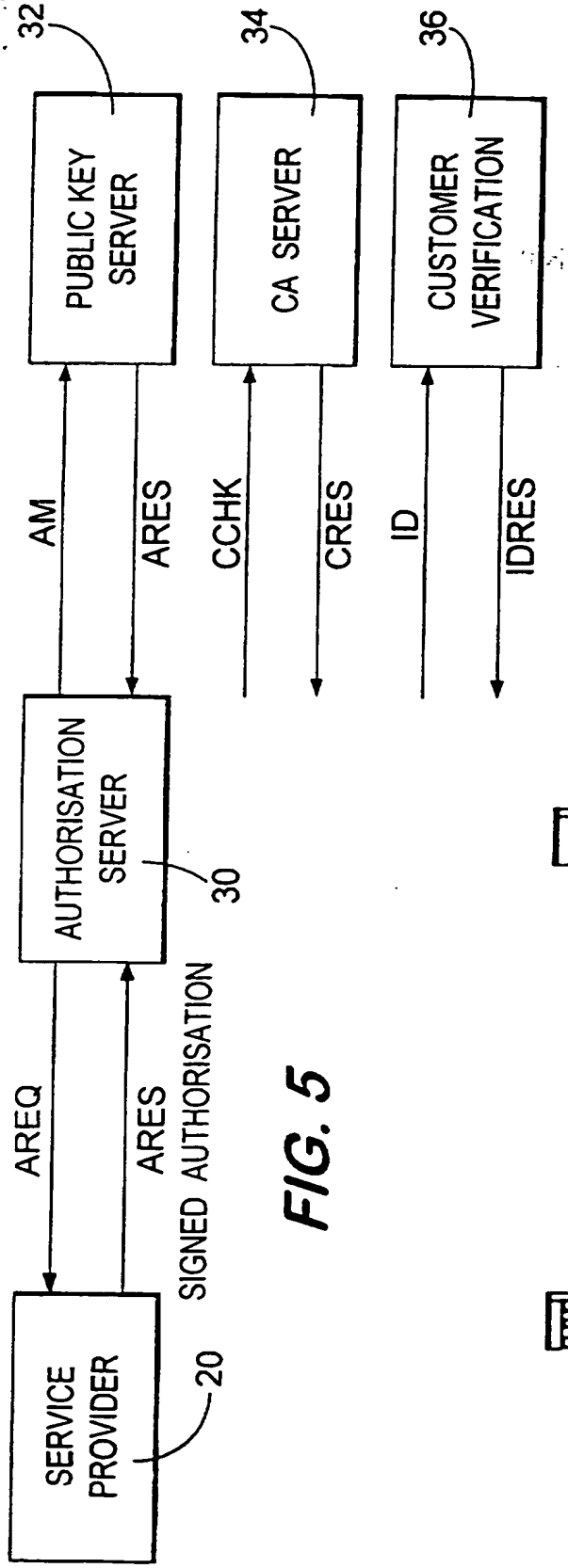


FIG. 5

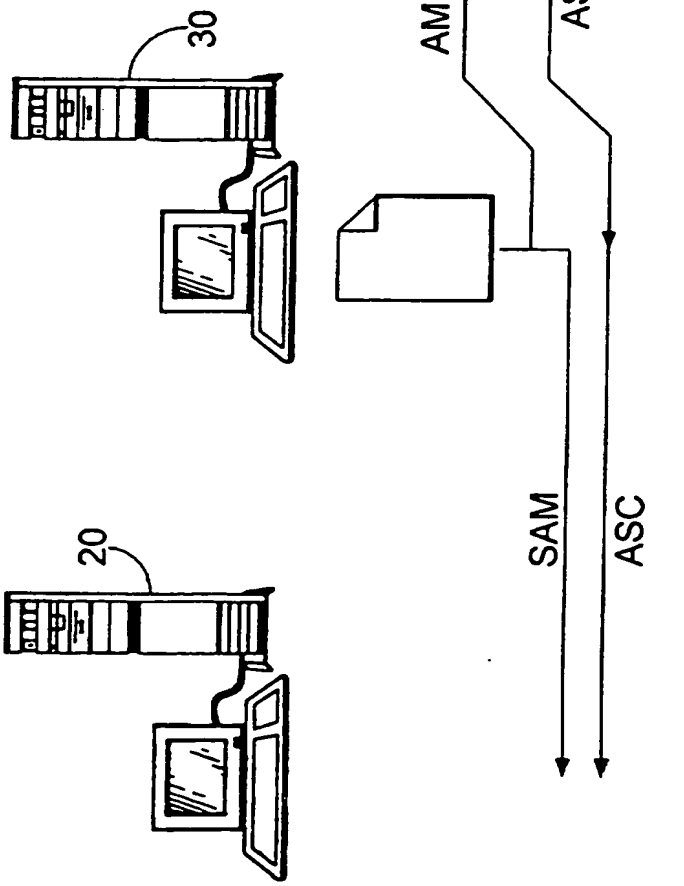


FIG. 13

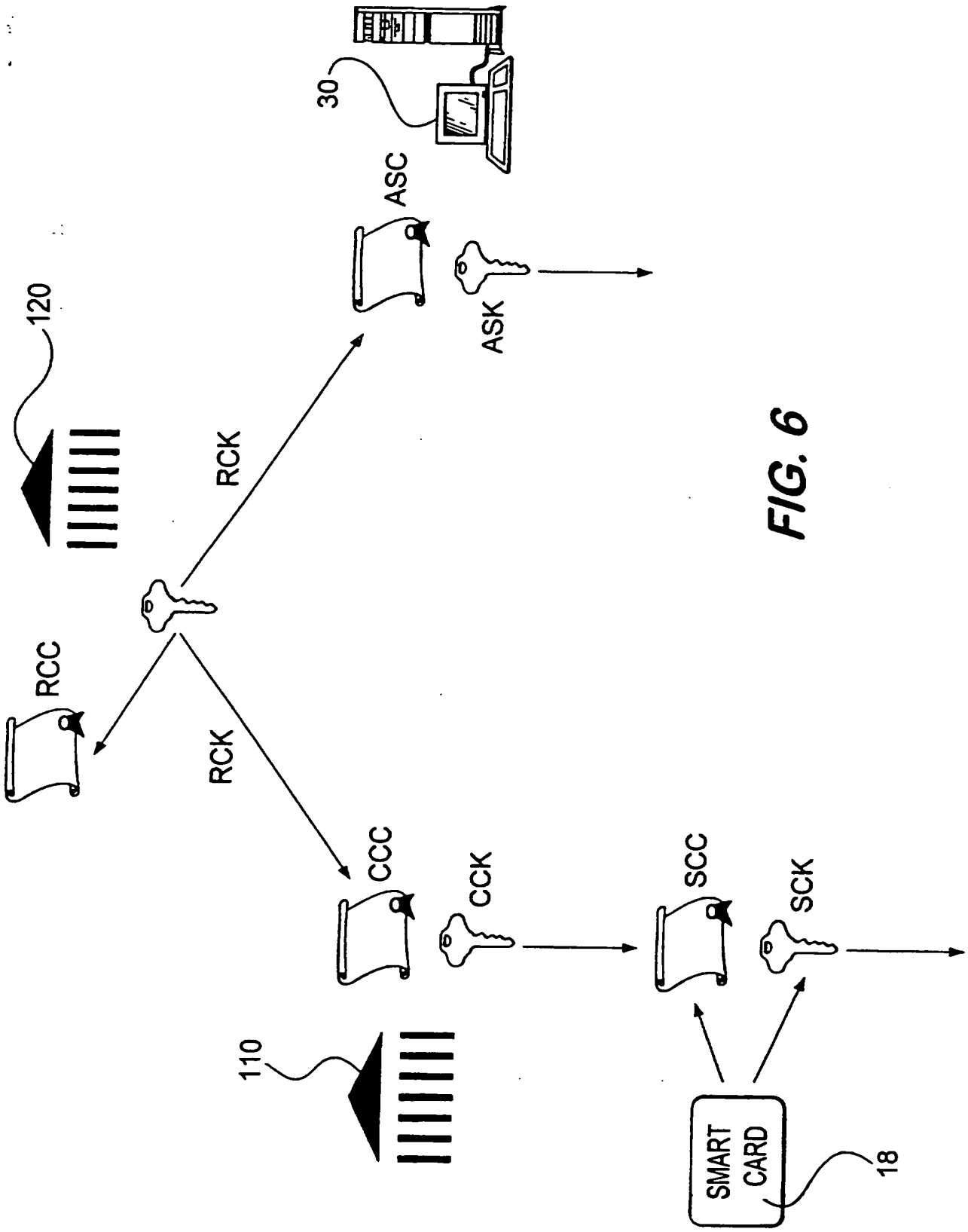


FIG. 6

FIG. 7

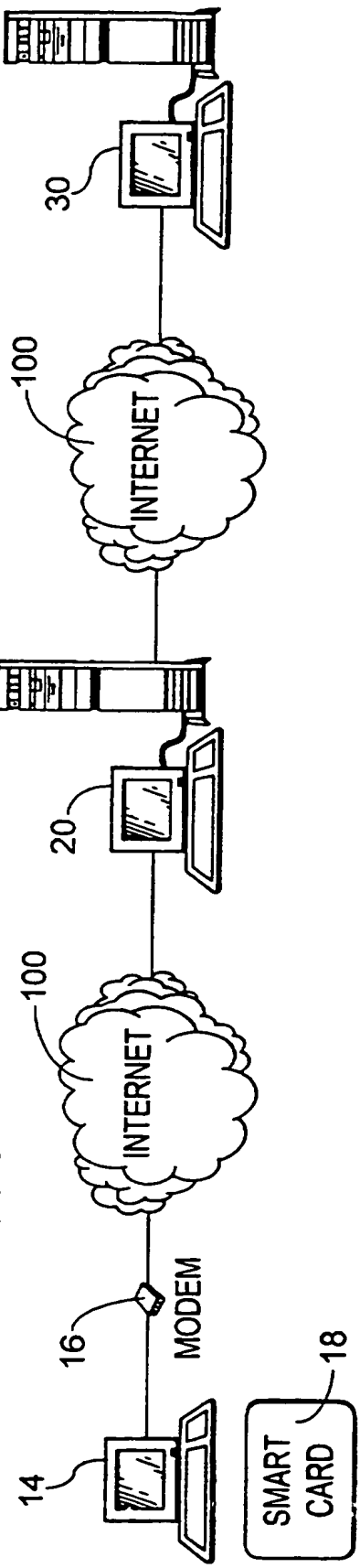
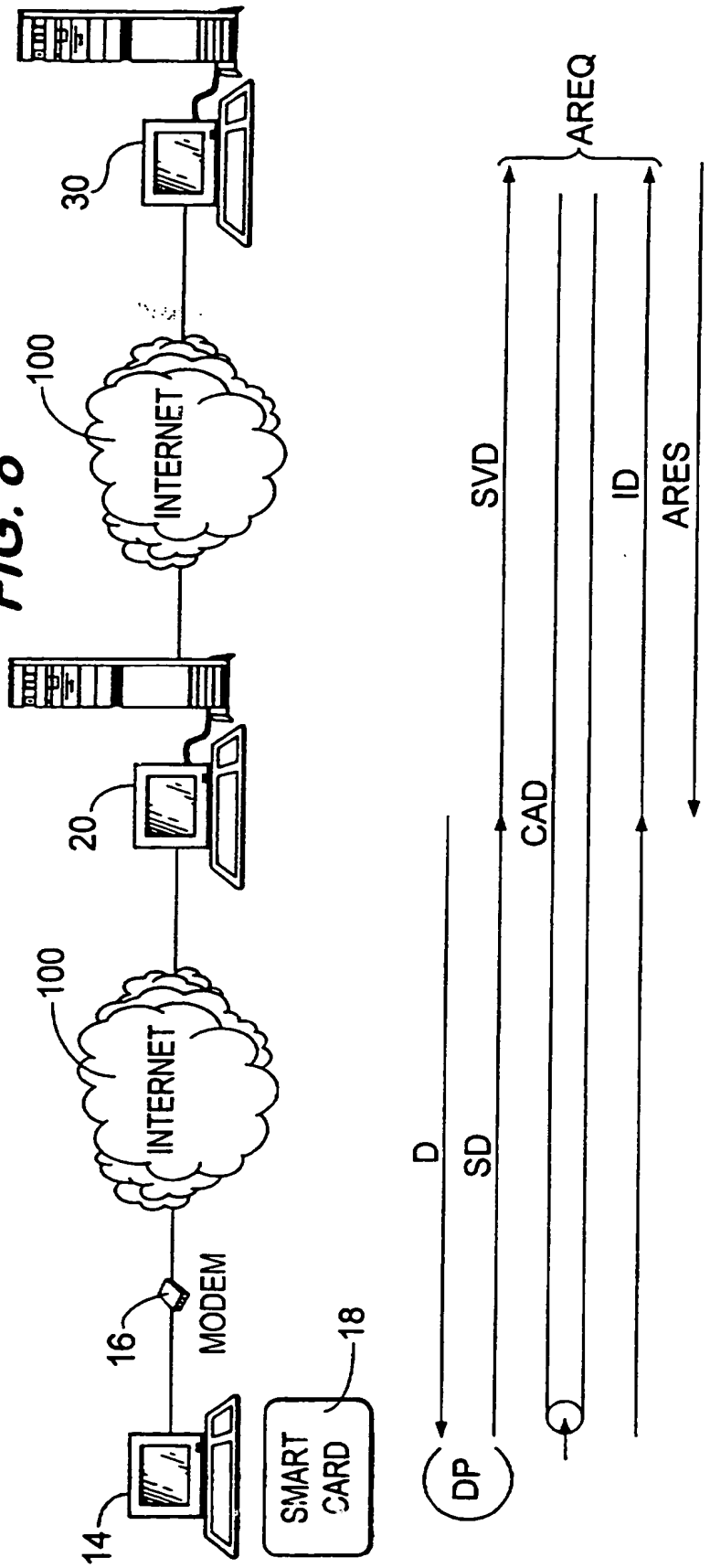


FIG. 8



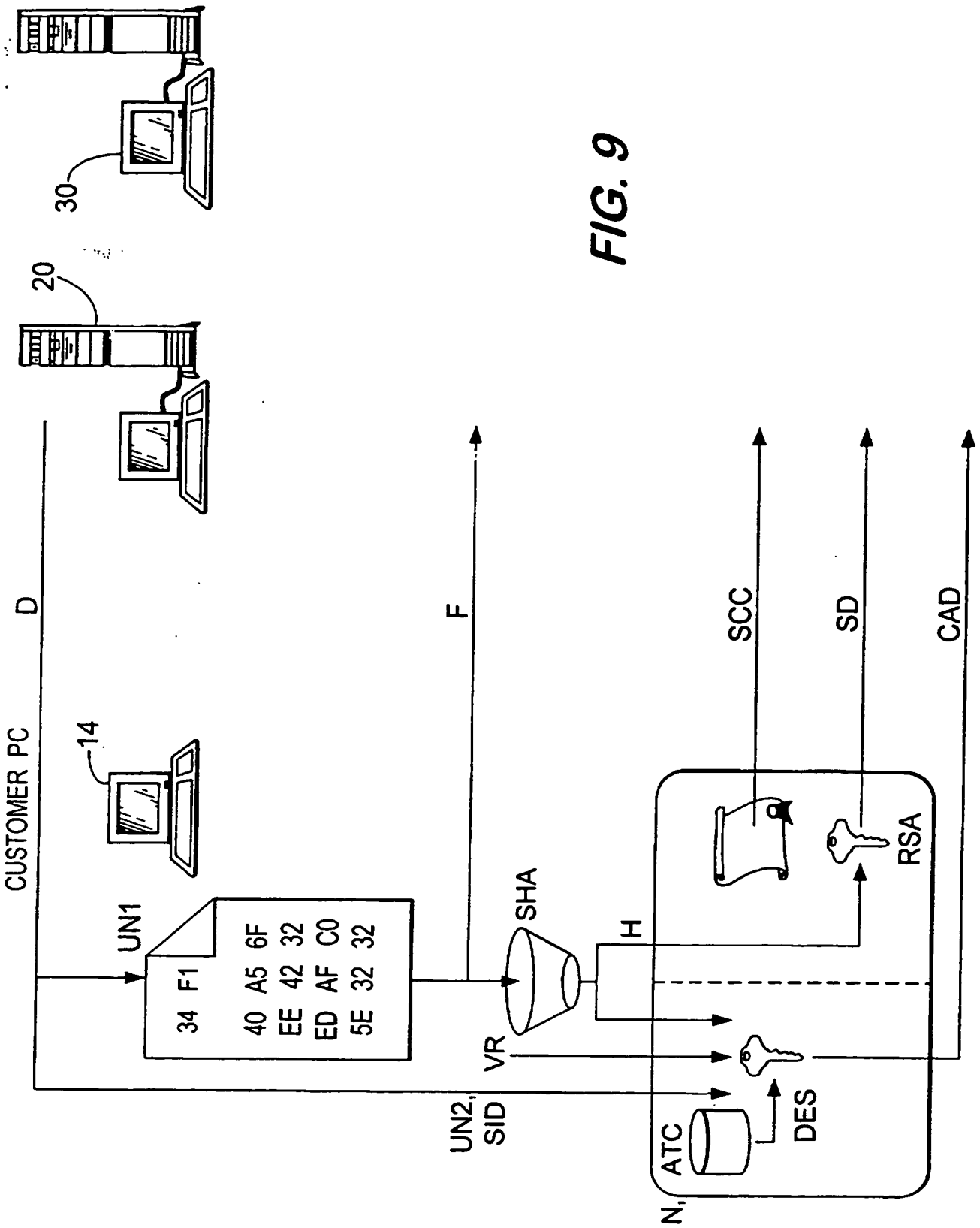


FIG. 9

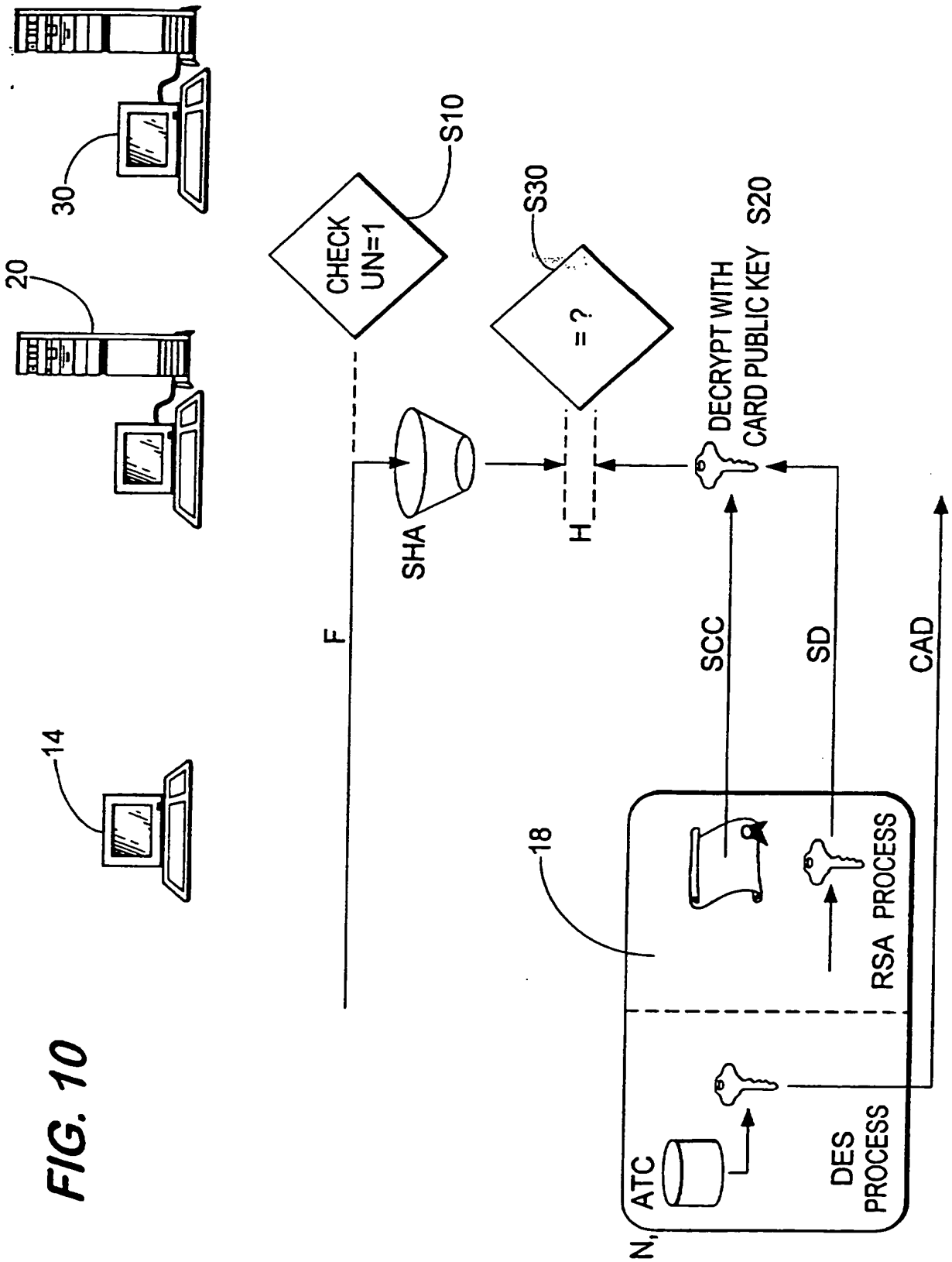


FIG. 10

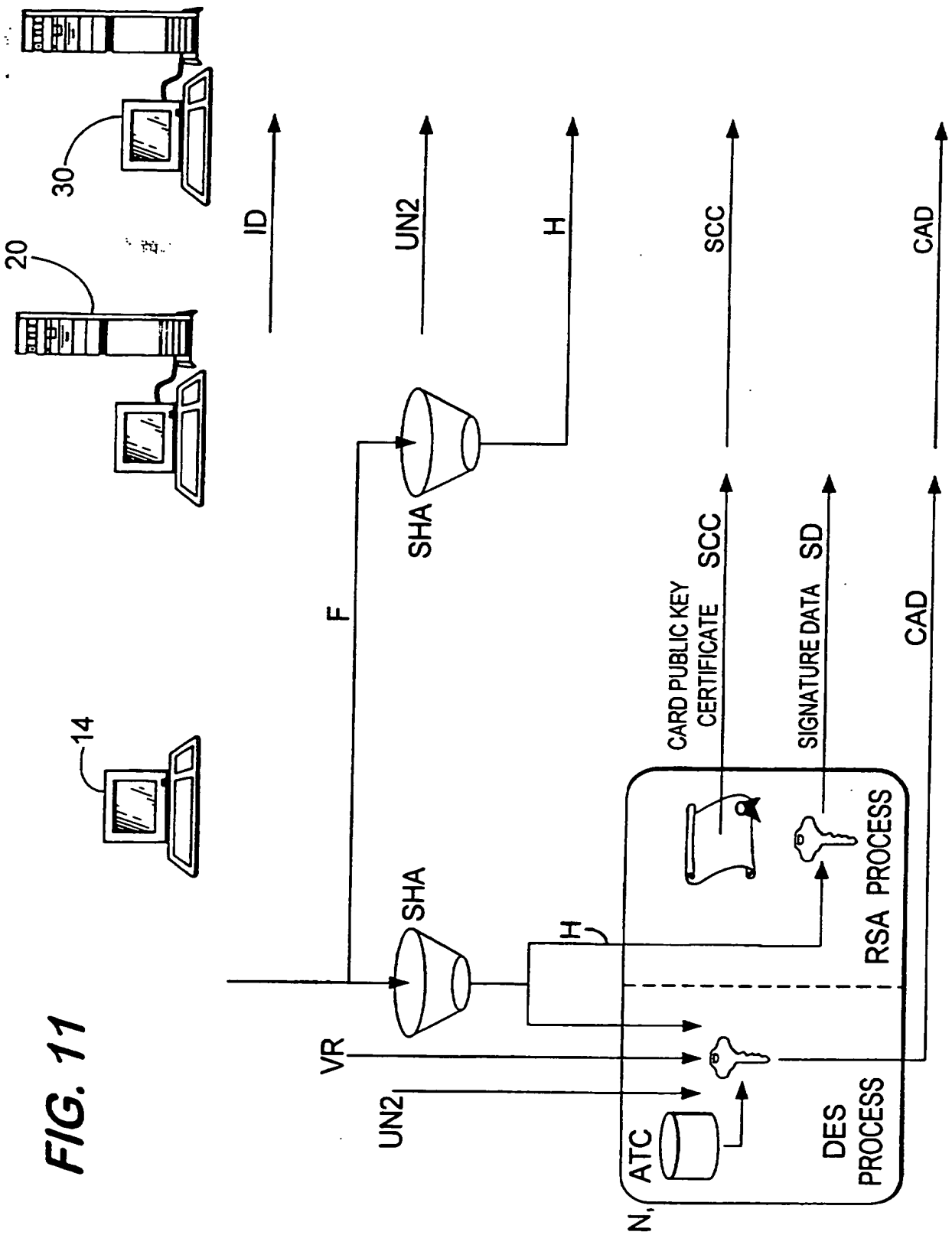
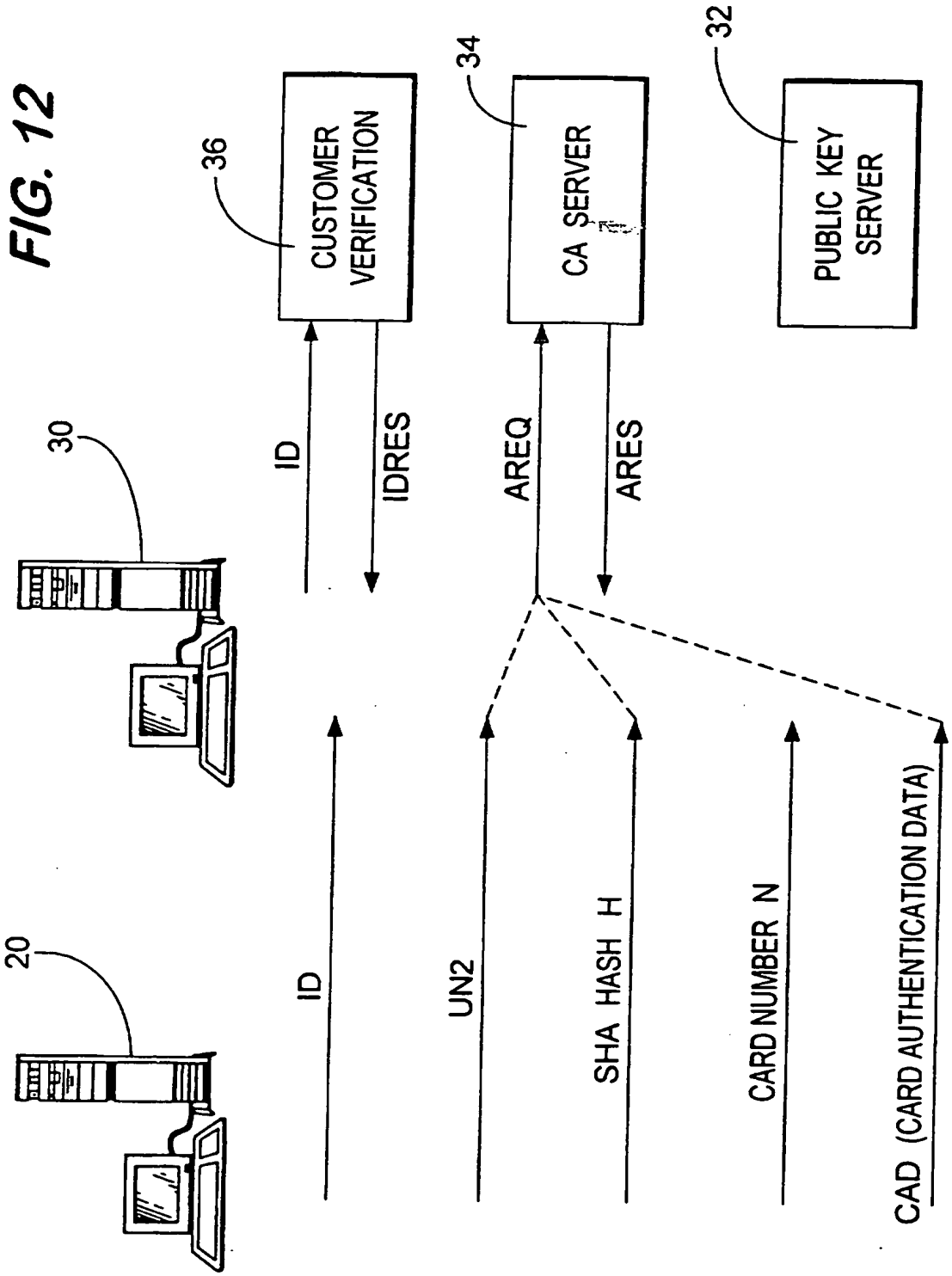


FIG. 11

FIG. 12



SECURE TRANSACTION SYSTEM

The present invention relates to a secure transaction system, particularly for use over a public network.

The most common and well-known public data network is the Internet, which provides network access to members of the public at low cost. Many types of commercial transaction which were previously conducted via telephone, mail or private network may now be conducted more easily over the Internet. However, internet protocols were not designed for security, and it has therefore been necessary to provide additional protocols for secure transactions over the Internet, including transport-level protocols such as the Secure Sockets Layer (SSL), and application-level protocols such as the Secure Hypertext Transfer Protocol (SHTTP). Such protocols aim to prevent interception, decryption and forgery of transactions between a client and a server over the Internet, but they do not verify the identity of the user of the client terminal. For example, in a credit card transaction, only the user's name and address, and the card number and expiry date need be supplied to order goods or services over the Internet. It is comparatively easy to obtain the necessary information to carry out fraudulent transactions over the Internet. Some verification of the user's identity is usually implemented at the application level, such as the use of passwords, but these too can be obtained or guessed.

Nevertheless, various protocols have been proposed for allowing secure transactions, and particularly secure payment, over the Internet; one example is Secure Electronic Transaction (SET), a protocol for credit card transactions over the Internet, a modification of which is described in WO 97/41539. Typically, such transactions involve three parties: a client which supplies credit card details, a service provider server operated by a supplier of goods or services, and an authorisation server which checks the credit card details and informs the service provider server whether payment is authorised by the operator of the credit card system.

However, conventional electronic transaction systems do not support non-repudiation; in other words, they do not provide sufficient evidence to confirm that a specific transaction was authorised.

Moreover, conventional electronic transactions do not bind a specific user to the use of an authorisation card.

Furthermore, conventional electronic transaction systems are limited in their application, because the authorising server is designed merely to give an authorise or decline message on the basis of the details supplied.

According to one aspect of the present invention, there is provided an electronic transaction system in which a terminal combines transaction data which is unique to a current transaction with terminal data which is unique to that terminal to generate bound terminal/transaction information which is sent to the transaction server. The transaction server sends the transaction data to

an authorisation server which returns information which binds the transaction to the identity of the authorisation server. The transaction server then has available information which binds together the transaction, the terminal and the authorisation server in a form which cannot be fraudulently created by the transaction server, and therefore cannot be repudiated.

According to another aspect of the present invention, there is provided an electronic transaction system in which an authorisation token is issued to a user. Information confirming the identity of both the authorisation token and the user are presented to an authority which confirms the validity of this information and makes it available to an authorisation server. The authorisation token is then used in an electronic transaction with a transaction server, in which user identification information and authorisation token information are supplied to the transaction server and passed to the authorisation server. The authorisation server compares the user identification information and authorisation token information with information previously made available by the authority and indicates to the transaction server the result of the comparison. In this way, because the correspondence between the user and the token has been verified before the transaction, the use of the token by the user can be confirmed during the transaction.

According to a further aspect of the present invention, there is provided an electronic transaction system in which a user terminal transmits to a transaction server transaction data and identification data. The transaction

data is passed from the transaction server to an authorisation server, which compares the identification data with stored identification data relating to authorised users of the system. The authorisation server then transmits to the transaction server an authorisation message indicating no authorisation, partial authorisation or full authorisation. In response to a partial authorisation message, the transaction server determines whether to accept the transaction data on the basis of the data content of the transaction data.

Specific embodiments of the present invention will now be described with reference to the accompanying drawings, in which:

Figure 1 is a diagram of the service functions of an authorisation system in an embodiment of the present invention;

Figure 2 is a diagram of the architecture of the system;

Figure 3 is a diagram of the hardware components of a user terminal;

Figure 4 is a diagram of the hardware components of a smartcard;

Figure 5 is a diagram of the sub-systems within the authorisation server in the embodiment;

Figure 6 is a diagram of the cryptographic architecture of the system;

Figure 7 is a diagram illustrating a more specific embodiment in which the system is used to authorise an electronic form;

Figure 8 shows the flow of data in the specific embodiment;

Figure 9 shows the cryptographic processes applied by the user terminal;

Figure 10 shows the digital signature validation process applied by the electronic form server;

Figure 11 shows the transmission of an authorisation request message to the authorisation server;

Figure 12 shows the authorisation process performed by the authorisation server; and

Figure 13 shows the authorisation response message process from the authorisation server to the electronic form server.

Authorisation Service

Figure 1 shows the service functions which provide a digital signature generation and authentication service in an embodiment of the present invention. A digital signature generator 10 generates digital signatures in the form of data which uniquely identifies a specific party and binds the signed data to that party. Digital signatures are a known technique for protecting data from modification and for identifying the signing party. The signing party is provided with a private/public key pair, which can be used to generate and verify digital signatures. The 'party' is usually assumed to be the user operating the computer which stores the private key and generates the digital signature. However, in the present embodiment, the private key is stored on a smart card and therefore the digital signature binds the signed data to the smart card. Hence, the 'party' is the smart card.

As is well-known, data encrypted with a private key can be decrypted with the corresponding public key, and vice versa. Suitable cryptographic algorithms are the RSA algorithm, described for example in US 4,405,829, and the Diffie-Helman algorithm described in US 4,200,770. In a typical digital signature process, the data to be 'signed' is subjected to a hash algorithm, such as the Secure Hash Algorithm (SHA). The result is a type of checksum, known as a 'hash', which depends both on the values of the bits making up the data and the positions of those bits. It is therefore very difficult to modify the data while giving the same hash. The hash is then encrypted using the private key, to produce a digital signature.

A digital signature can be verified by performing the same hash algorithm on the received data as was used to generate the hash encoded in the digital signature. The digital signature is then decrypted using the public key and the decrypted hash is then compared to the calculated hash. If the hashes match, the signature is valid.

Examples of digital signature algorithms are the RSA digital signature, in which the hash is encrypted together with an indication of the type of the hash algorithm using RSA encryption, and the digital signature algorithm (DSA), in which the hash algorithm is the SHA and the hash is encrypted together with a random number by a variant of the Diffie-Helman algorithm.

The digital signature generator 10 exchanges information IE with a digital signature acceptor 20, and sends 'signed' information SI to the digital

signature acceptor which includes a digital signature generated using a private key held by the digital signature generator. On receipt of the 'signed' data, the digital signature acceptor 20 sends an authorisation request AREQ to a digital signature authoriser 30. The authorisation request AREQ includes information derived from the signed information SI. The digital signature authoriser 30 checks the information derived from the signed information SI, determines whether the digital signature should be accepted as genuine, and sends to the digital signature acceptor an authorisation response ARES indicating whether the signature is to be accepted as genuine. The authorisation response ARES is signed with a digital signature generated by a private key held by the digital signature authoriser 30. Dependent on the authorisation response, the digital signature acceptor then accepts or rejects the signed information SI.

In order to provide evidence of the provision and acceptance or rejection of the signed information, the digital signature acceptor 20 may further transmit a notarisation request NREQ to a signed message notariser 40, including information derived from the signed information SI and from the authorisation response ARES. In response, the signed message notariser 40 transmits to the digital signature acceptor 20 a notarisation confirmation NCF which includes information derived from the notarisation request NREQ, digitally signed by the signed message notariser 40. The digital signature acceptor 20 transmits archive deposit information ARDEP, which may for example comprise the signed information SI, the authorisation response

ARES and the notarisation confirmation NCF, to a long-term archive 50. This information is later retrieved from the long-term archive 50 as archive retrieval information ARRET in the event of repudiation of the signed information SI.

Optionally, where an independent time reference is needed, for example to confirm the exact time of a transaction such as the sending and receipt of the signed information SI, information from a standard time signal source 60 is made available to each of the functions described above and is incorporated in each digitally signed message.

System Architecture

A high-level system architecture of the digital signature authorisation service is shown in Figure 2. Like parts to those of Figure 1 carry the same reference numerals.

A user 12 wishing to subscribe to the digital signature service must first apply for a smart card 18 from which the user's digital signatures are derived. The user 12 has a computer 14 connectable to the Internet 100 via a modem 16, using for example web browser software and TCP/IP protocols as is well-known in the art.

The components of the computer 14, which may be an IBM-compatible 'personal computer' or Apple Macintosh computer, are shown in Figure 3. A CPU 110 is connected through a bus 120 to main memory 130, a

disc controller 140 connected to a hard disc 145, a user interface controller 150 connected to a keyboard 152 and other input devices such as a mouse 154, a display controller 160 connected to a visual display 165, and an I/O controller 170. The I/O controller 170 controls the modem 16 and a card reader 174 into which a smart card can be docked. Optionally, a biometric device 180 is connected to the I/O controller 170 or to the user interface controller 150. The biometric device 180 may be a fingerprint scanner, an iris scanner, or another device which allows information derived uniquely from the user 12 to be input to the computer 14. The fingerprint scanner may be integrated with the keyboard 152.

The user accesses a customer server 70 through the Internet 100 and requests subscription to the digital signature service. The request is passed on to a card bureau 90 by a suitable form of secure communication. The card bureau 90 sends a smart card 18 to the user 12. An example of the components within the smart card 18 is shown in Figure 4.

The smart card 18 contains a processor 200 connected to a memory 210 and an external connector 220. The card 18 may include a power source such as a cell integrated within the card 18, or power may be supplied through the connector 220 by the card reader 174. At least part of the memory 210 is non-volatile so that a operating program and data are stored when the card 18 is removed from the reader 174.

A public/private key pair, a card identity code and a PIN are recorded in the non-volatile memory of the card 18 during manufacture. When a request is received from the customer server 70, the name of the user 12 is printed on the card 18, which is then sent to the user 12. A status message is sent to the authorisation server to indicate that the card 18 is issued but inactive.

The user 12 then takes the card 18 to the public premises 80 of an organisation which supports the digital signature service, where the user's identity is checked against the identity of the user to whom the card 18 has been issued. Once the user's identity has been verified, a status message is sent from the premises 80 to the authorisation server 30 identifying the card 18 and the user 12. The user 12 is also issued with the card reader 174 and application software for the digital signature service, if the user 12 does not already have these. Subsequently, the user 12 is notified of the PIN.

To use the authorisation service, the user 12 inserts the card 18 into the card reader 174. Application software running on the computer 14 then prompts the user 12 for a PIN. The user enters a PIN which is compared to that stored on the card 18, and the application generates a card validation result (VR) which indicates whether a PIN has been requested and whether the entered PIN matched that stored on the card 18.

Additionally or alternatively, the computer 14 obtains biometric data from the biometric device 180 if this is present. The smartcard 18 generates a

digital signature for the signed information SI transmitted by the computer 14 to the service provider 20, as will be described in a specific example below.

The service provider 20 may comprise a general purpose computer running web server software and connected to the Internet 100. The service provider 20 also runs authorisation software for communication with the authorisation server 30.

The authorisation server 30 comprises a general purpose computer running authorisation server software and connected to the Internet 100. As shown in Figures 2 and 5, the authorisation server 30 is also connected, for example over a local or private network, to a public key server 32, a card authentication server 34 and a customer verification server 36. The public key server 32 and the card authentication server 34 comprise dedicated hardware modules including encryption /decryption acceleration hardware. The customer verification server 36 stores a database containing the details of authorised users of the digital signature service.

The authorisation server 30 receives from the service provider 20 the authorisation request AREQ, which includes a hash of the signed information SI, a public key certificate, identification information relating to the card 18 and the user 12, and card authentication information. The authorisation server 30 sends the card authentication information CCHK to the card authentication server 34, which checks the authenticity of the card 18 and returns a response CRES indicating whether the card is authentic or not. The authentication

server 30 sends the user identification information ID to the customer verification server 36, which returns a response IDRES indicating whether, or to what extent, the customer identity details are correct.

The authorisation server 30 generates from the responses CRES and IDRES an authorisation message AM, which is sent to the public key server 32. The public key server 32 signs the authorisation message AM to produce a signed authorisation response ARES which is transmitted to the service provider 20.

Public Key Hierarchy

The digital signatures are generated, verified and authorised using public key cryptography, as explained above. The public keys are distributed by means of public key certificates, so that users of public keys can trust that the public keys belong to the parties with which the users wish to communicate. As is well known, a public key certificate consists of a name identifying a party who holds a private key (in this case, the card 18), the corresponding public key, and a digital signature comprising a hash of the name and the public key, encrypted using the private key of a certification authority trusted by the user. If the user does not have the certification authority's public key, this can be obtained from the certification authority's public key certificate, which is signed by a root certification authority. Thus, a

hierarchy of public key certificates may be used, ultimately administered by a root certification authority which is always trusted.

If a public/private key pair is changed, however, the public key certificate is no longer valid. The old public key certificate may be revoked by placing a code identifying that certificate on a Certificate Revocation List CRL, which is circulated periodically to users of the public key. Before a public key certificate is used, it may first be checked against the CRL.

Figure 6 shows the hierarchy of keys in the present embodiment. The smartcard 18 performs the digital signature process using a private key SCK stored within the card 18. The corresponding public key is contained within a smart card certificate SCC stored within the card and signed using a card certification private key CCK of a card certification authority 110. The corresponding public key is contained within a card certification authority certificate CCC which is signed by a root certification authority private key RCK of a root certification authority 120. The root certification authority private key RCK is also used to sign the public key certificate ASC of the authorisation server 30. The corresponding public key is distributed in a root certification authority public key certificate RCC, which is self-signed using the corresponding private key RCK. The authorisation server private key ASK is used to provide a digital signature on the authorisation response ARES.

Card Authentication

In addition to the public key transactions described above, the smart card 18 also performs a symmetric key card authentication function with the authentication server 30, using a separate private key stored on the card 18. The process uses a two-key triple data encryption standard (DES, encrypt-decrypt-encrypt) algorithm in Cipher Block Chaining Mode to produce an eight byte message authentication code (MAC).

The symmetric key authentication function is used for secure communications between the smartcard 18 and the authentication server 30, which the service provider 20 passes transparently but is unable to decrypt. For each card authentication transaction, the MAC is calculated from a combination of: data stored internally in the card 18, including a variable application transaction counter value ATC which is incremented for each transaction, and a card identity number N which is included in the public key certificate SCC; data supplied by the service provider 20, including the time, date and the identity of the service provider, so as to bind the MAC to the specific transaction; a hash of the signed information SI, so as to bind the MAC to the data supplied and to the digital signature; and the validation result VR or information derived from the biometric data.

Electronic Forms Implementation

A more specific embodiment will now be described with reference to Figures 7 to 13, in which the digital signature system is used to authenticate a

completed form supplied electronically by the user 12 to the service provider 20, which is in this case operated by a government department. For convenience, Figures 7 to 13 show the topological connection between the computer 14 and the service provider 20, and between the service provider 20 and the authorisation server 30, as being through separate networks but in this example the connections are both through the Internet.

Figure 8 shows the data transmission which takes place during a transaction. The computer 14 has already established an Internet connection to the service provider 20, and is running web browser software. In response to a request initiated by the user 12, the service provider 20 sends data D comprising HTML pages representing a blank form (such as a form for registering a new business). The user 12 completes the form by entering data within the browser software and requesting submission of the completed form to the service provider 20. The browser software supports the digital signature service, for example by means of a 'plug-in' which modifies standard browser software, so that the user 12 is prompted to enter a PIN when requesting submission of the completed form. If the smartcard 18 is not located in the card reader 174, the software prompts the user 12 to do this.

At a data processing stage DP, the smartcard 18 generates a digital signature from the form data F of the completed form and the smart card private key SCK. The signed data SD is then sent to the service provider 20, which verifies the signature by recovering the card public key from the smart

card certificate SCC using the card certification public key recovered from the card certification authority certificate CCC, recalculating the hash function for the form data and comparing the recalculated hash function with the one signed with the smart card key SCK and included in the signed data SD. Signature verification data SVD derived from the signed data SD is sent by the service provider 20 to the authorisation server 30.

Card authentication data CAD, which comprises the MAC and the input data used to generate the MAC, and the user identification data ID are transmitted to the service provider 20 and passed to the authentication server 30. The user identification data ID comprises for example the name, address and date of birth of the user, entered by the user 12 and transmitted by the browser software in a separate field from the signed data SD. Optionally, biometric data from the biometric device 180 is included in the user identification information. Collectively, the signature verification data SVD, the card authentication data CAD and the user identification data ID comprise the authorisation request AREQ submitted to the authorisation server 30.

The authorisation server 30 checks the authorisation request AREQ for counterfeit or replayed cryptograms, checks whether the smartcard public key certificate SCC matches the card identity and checks the user identification data ID against an entry in a database of details of known cardholders. The results of these checks are digitally signed using the authorisation server

private key ASK and sent as the authorisation response ARES to the service provider 20.

Some of the above processes will now be explained in greater detail.

User to Service Provider

Figure 9 shows how the data sent from the user's computer 14 to the service provider 20 is generated. The data D sent from the service provider 20 includes two random or otherwise unpredictable 32-bit numbers UN1 and UN2, the date and time of the transaction and a server identifier SID. The first unpredictable number is embedded in the HTML form as a readable serial number and is returned in the form data F.

The application software calculates a hash of the completed form data F using a secure hash algorithm SHA and sends the hash to the card 18, together with the second unpredictable number UN2, validation result VR, the date and time and the server identifier SID, for use in the DES encryption process to generate the MAC. The card generates an application transaction counter value ATC which is also input to the DES process. The hash is also supplied as an input to an RSA public key encryption process. The card public key certificate SCC is stored in the card 18 and is retrieved by the application software together with the MAC and signature data SD, for transmission to the service provider 20.

Signature Validation

The process performed by the service provider 20 to validate the signature of the signed data SD is shown in Figure 10. The service provider 20 receives the form data F and checks (S10) that the serial number UN1 matches that of the blank form previously sent. A hash is calculated from the form data F using the same SHA as performed by the card 18. The card public key is retrieved from the card public key certificate SCC and is used to decrypt (S20) the signature data SD to extract the hash calculated by the user's computer 14. The two hashes are compared (S30) and the signature is validated if they match. However, this process only establishes that the form was digitally signed using the card 18. The service provider 20 must further check that the card 18 is valid and is being used by the authorised cardholder, as will be described below.

Authorisation Request

The service provider 20 sends the following information to the authorisation server 30 in the authorisation request message ARES, as illustrated in Figure 11: the user identification data ID, including any biometric data, the second unpredictable number UN2, the hash H calculated from the received form data F, the card public key certificate SCC and the message authentication code MAC. The form of the authorisation request message is shown in detail in Table 1 below:

Table 1 - Authorisation Request Message

Field Name	Field Description
Version	Version number of the cryptogram
Service Provider Ref.	Message reference supplied by service provider
Authorisation Service Ref.	Authorisation service reference supplied by the service provider
Contract Info	A URL for a contract governing the use of the authorisation service
Request Sent Time	The time, as supplied by the service provider system clock, at which the message was transmitted
SCC	Public key certificate of the card 18
H	Hash of the form data F
User Time	The time, from the user's computer 14, at which the authorisation request was transmitted from the computer 14
Service Provider Identity	Service provider identity used for generation of the MAC
UN2	Second unpredictable number
Card Data	Data generated by the application and the card 18 and passed to the authorisation server

User Surname	User's surname
User First Name	User's first name(s)
User Title	User's title
User DOB	User's date of birth
User Address	First line of user's address
User Postcode	User's postcode

The Card Data described in Table 1 comprises the fields shown below in Table 2:

Table 2 - Card Data Structure

Field Name	**Field Description**
Cryptogram Info Data	Indicates the type of the cryptogram
Application Transaction Counter	A counter value, stored in the card 18, which is updated after each transaction
MAC	Cryptogram generated by the card 18
Issuer Application Data	Data determined by the card issuer

Authorisation Request Process

The authorisation server 30 determines the authorisation response ARES as illustrated in Figure 12. The user identity information ID is sent to

the customer verification server 36, which checks whether this information matches stored user identity information related to the card number, and returns a response IDRES indicating whether these details are correct and the status of the card. The card authentication data CAD are sent to the card authentication server 34 where the MAC is verified using the card number N, extracted from the card public key certificate, the second unpredictable number UN2 and the date, time and server identity originally provided by the service provider 20. The card authentication server 34 also extracts the validation result VR, determines whether the cryptogram has been replayed or the card counterfeited, and whether the card public key certificate SCC is valid and corresponds to the MAC.

Optionally, the customer verification server 36 stores a database of biometric information for each authorised user, and the response IDRES includes information on the confidence level with which biometric data contained within the user identity information ID matches the stored profile for that user.

Authorisation Response

The authorisation server 30 formats the authorisation response message ARES and transmits it to the service provider 20 by means of a process illustrated in Figure 13. First, an authorisation message AM is generated including the following information: the hash value H and the user

identification information ID copied from the authorisation request AREQ, and a response code indicating the card authentication and user verification server responses. This message AM is sent to the public key server 32 which generates a digital signature for the message using the authorisation server private key ASK and returns this signature AS together with the authorisation server public key certificate ASC. The authorisation server then sends the authorisation message AM, the signature AS and the authorisation server certificate ASC to the service provider 20, together with a reference code indicating the agreement under which the authorisation is made to the service provider 20.

The data content of the authorisation message is summarised below in Table 3:

Table 3 - Authorisation Message Contents

Field Name	**Field Description**
Service Provider Reference	Message reference number supplied by the service provider, copied from the authorisation request message
Hash	Hash of the authorisation request message
Request Received Time	The time, from the authorisation server, at which the authorisation request was received
Authorisation Response	Authorisation Response Data
Response Sent Time	Time, from the authorisation server, at which

	the response was sent

The Authorisation Response Data is coded as individual bits, as shown in Table 4:

Table 4 - Authorisation Response Data Bit Meaning

Index	Bit No.	Meaning
0	0	Card does not exist
	1	Card not activated
	2	Card expired
	3	Card reported lost
	4	Card reported stolen
	5	Could not verify
	6	Card registered as demo
	7	Verification error - see following bytes
1	0	Unspecified address match failure
	1	Surname did not match
	2	First name did not match
	3	Title did not match
	4	Date of birth did not match
	5	First line of address did not match
	6	Postcode did not match

	7	(Unused)
2	0	Unspecified authentication failure
	1	Card authentication could not be performed
	2	Cryptogram verification failure
	3	Application Transaction Counter value invalid
	4	PIN verification not performed
	5	PIN verification failed

Optionally, the authorisation response data may include an indication of the confidence level with which the biometric data included in the customer identification information matches that stored in the customer verification server 36; for example, pattern matching algorithms used in iris and fingerprint scans will return a suitable value based on the correlation between an input and a reference pattern.

On the basis of the authorisation response ARES, and the result of the signature verification process, the service provider 20 determines how the form data F is to be processed. For example, if the signature is verified and the transaction authorised by the authorisation server, the service provider may update a record corresponding to the user 12 to add the information included in the form data F. If either the signature is not verified, or the transaction not authorised, the form data F may be discarded and/or a message transmitted to the user's computer indicating that the form has not been accepted.

Since the authorisation response ARES is able to indicate the reason for a negative authorisation, the service provider may allow certain types of transaction to proceed if an insignificant authorisation failure is indicated. For example, if the title did not match, the service provider judges this as insignificant, since the user is unambiguously identified by other data, and the transaction is processed as if a positive authorisation were issued by the authorisation server 30.

In the option where the authorisation response ARES includes an indication of the confidence level with which the supplied user identification information ID matches the stored user identification information, the service provider 20 sets a minimum confidence level for the current transaction and allows the transaction to proceed if this confidence level is exceeded. Preferably, this minimum confidence level is determined according to the monetary value of the transaction, or if the transaction does not have a specified monetary value, the consequences of fraud in that transaction.

Preferably, the authorisation response ARES is used to determine the liability in the case of fraud between the operator of the service provider 20 and the operator of the authorisation server 30. For example, if any of the bits with index 0 are set, the card system operator will not accept any liability if the service provider accepts the transaction, but if only one of the bits with index 1 is set, the card system operator will accept liability only to a

prearranged limit, and if none of the bits is set, the card system operator will accept liability to the maximum value prearranged for the user 12.

The present invention is not limited to the use of the Internet but may also be applied to transactions over other networks which are not inherently secure. The network used to connect the user's computer 14 to the service provider 20 may be separate from that used to connect the service provider 20 to the authorisation server 30.

Although the above embodiment is described with reference to one specific user, it is evident that similar procedures are carried out for different users so that the system can perform transactions initiated by any one of a large number of users.

The present invention is not limited to any specific type of transaction such as the authentication of forms or the authorisation of payment.

In an alternative embodiment, the smart card 18 may sign the form data using the symmetric encryption key under the DES process and may dispense with the RSA encryption process. The service provider 20 will not then be able to verify the signature, but instead recalculates the hash and transmits this to the authorisation server 30 for verification.

In an alternative embodiment, the smart card 18 uses the private key SCK to produce the MAC and does not use any symmetric encryption key. In that case, the service provider 20 can verify the authenticity of the MAC.

The above embodiments are described by way of example and are not to be construed as limiting the scope of the invention. Instead, the invention extends to all variants which fall within the scope of the following claims.

CLAIMS

1. A method of processing a data transaction between a terminal and a first server over a public network and between the first server and a second server, comprising:

generating, at the terminal, terminal encrypted data derived from transaction identification information uniquely identifying a current transaction, using a first key;

transmitting said terminal encrypted data from said terminal to said first server over said public network;

transmitting said terminal encrypted data and said transaction identification information from said first server to said second server;

decrypting said terminal encrypted data using a second key;

comparing said decrypted data with said transaction identification information received from said first server by said second server;

and transmitting a transaction authorisation message from said second server to said first server, the content of the authorisation message being dependent on the result of said comparison.

2. A method as claimed in claim 1, including inputting user identification information from a user at said terminal,

wherein the user identification information is transmitted together with said terminal encrypted data to said first server and from said first server to said second server, the method further comprising:

comparing the received user identification information at said second server with previously stored user identification information, the content of the authorisation message being dependent on said comparison of said user identification information.

3. A method as claimed in claim 1 or claim 2, wherein said first and second keys are symmetric encryption keys.

4. A method as claimed in any preceding claim, wherein said authorisation message is signed by the second server using a third key.

5. A method as claimed in any preceding claim, further comprising:

inputting transaction message data at said terminal by said user;

signing said transaction message data at said terminal using a fourth key to generate transaction signature data;

transmitting said transaction message data and said transaction signature data from said terminal to said first server; and

verifying said transaction signature data against said transaction message data using a fifth key at said first server.

6. A method as claimed in any preceding claim, including supplying said authorisation message and transaction identification information from said first server to data storage means.

7. A method of processing a data transaction between a terminal and a first server over a public network and between the first server and a second server, comprising:

transmitting transaction message data and identification data from said terminal to said first server;

transmitting said identification data from said first server to said second server;

comparing said received identification data at said second server with previously stored identification data and generating an authorisation message indicating either no authorisation, partial authorisation or full authorisation dependent on said comparison;

transmitting said authorisation message from said second server to said first server;

and processing said transaction data at said first server according to the received authorisation message.

8. A method as claimed in claim 7, wherein said transaction data is processed at said first server further according to the content of the transaction data.

9. A method as claimed in claim 8, wherein the processing step comprises:

determining a threshold value from the authorisation message;

determining a transaction value from the transaction data;

and processing the transaction data as valid if the transaction value does not exceed the threshold value.

10. A method as claimed in any one of claims 7 to 9, wherein the authorisation message is signed by the second server using a first key, and the signed authorisation message is verified by the first server using a second key, the processing of the transaction data being dependent on said verification by the first server.

11. A method of authenticating a data transaction between a user terminal and a first server over a public network, comprising:

issuing an authentication token to a user, said authentication token storing authentication information;

verifying the identity of the user and of the token and transmitting a verification message to a second server;

storing status information at said second server in response to said verification message;

inputting user identification information and transaction data at a user terminal;

connecting said authentication token to said terminal and retrieving said authentication information therefrom;

transmitting said user identification information, authentication information and transaction data from the user terminal over a public network to the first server;

transmitting said user identification information and authentication information from said first server to said second server;

comparing said received user identification data with previously stored identification data;

retrieving said status information corresponding to said authentication information;

and transmitting an authorisation message dependent on the result of said comparison from the second server to the first server and on said status information.

12. A method as claimed in any preceding claim, wherein said transmissions between the first server and the second server are performed over said public network.

13. A method as claimed in any preceding claim, wherein said public network is an internet.

14. A method as performed by the first server in a method as claimed in any preceding claim.

15. A method as performed by the second server in a method as claimed in any one of claims 1 to 13.

16. Apparatus arranged to perform the method as claimed in claim 14.

17. Apparatus arranged to perform the method as claimed in claim 15.

18. A method substantially as herein described with reference to the accompanying drawings.

The Patent Office

34

Application No: GB 9812520.6
Claims searched: 1-18

Examiner: B.J.SPEAR
Date of search: 10 December 1998

Patents Act 1977
Search Report under Section 17

Databases searched:

UK Patent Office collections, including GB, EP, WO & US patent specifications, in:

UK Cl (Ed.P): H4P(PDCSA,PDCSC)

Int Cl (Ed.6): H04L 9/32

Other: Online:WPI,EPODOC,PAJ

Documents considered to be relevant:

Category	Identity of document and relevant passage		Relevant to claims
X	EP0782296A2	(NCR) Whole document, eg Fig.8 and p 2 l 50 to p 3 l 45 p 5 l 37-48.	1-3,7,11-17 at least
X	WO97/41539A1	(Verifone) Whole document, eg Figs. 1B, 4,6A,6B, abstract and p 19 l 39 to p 24 l 17.	1-3,7,11-17 at least
X	WO97/23972A1	(V-ONE) Whole document, eg Fig. 1,abstract, and p 3 l 18 to p 4 l 8, p 7 l 2 to p 8 l 26, p 12 l 8-19	1-3,7,11-17 at least
X	US5615268	(Document Authentication) Whole document, eg Fig.3 and col. 6 l 11 to col. 7 l 33.	1-3,7,11-17 at least
X	US5602918	(Virtual Open) Whole document, eg Fig. 1 and col. 3 l 31 to col. 5 l 44	1-3,7,11-17 at least
X	US5590197	(V-One) Whole document, eg Fig. 1 and col. 4 l 42 to col. 7 l 8.	1-3,7,11-17 at least
X	Internet Cryptography by Richard E Smith Pub 1997 Addison Wesley ISBN 0-201-92480-3 eg pages 113-116, 262-265		1-3,7,11-17 at least

X Document indicating lack of novelty or inventive step
Y Document indicating lack of inventive step if combined with one or more other documents of same category.
& Member of the same patent family
A Document indicating technological background and/or state of the art.
P Document published on or after the declared priority date but before the filing date of this invention.
E Patent document published on or after, but with priority date earlier than, the filing date of this application.